

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-262042

(43)Date of publication of application : 13.10.1995

(51)Int.Cl.

G06F 11/22

(21)Application number : 06-046245

(71)Applicant : FUJITSU LTD

(22)Date of filing : 17.03.1994

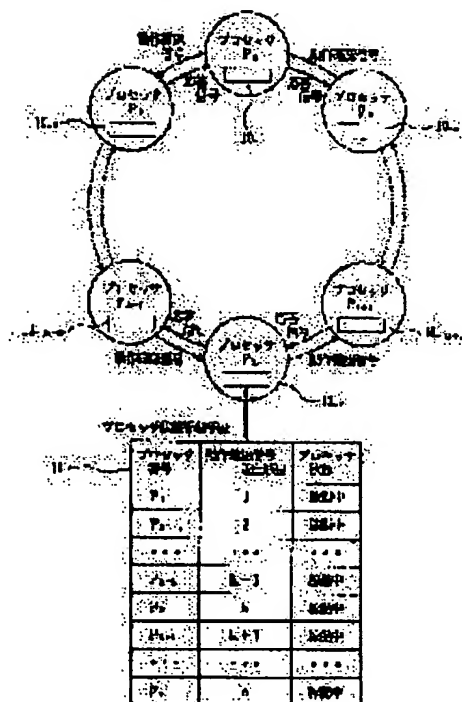
(72)Inventor : JO JUNKO

(54) DETECTING METHOD FOR FAULT OF PROCESSOR

(57)Abstract:

PURPOSE: To provide a fault detecting method reducing the processing burden on processors concerning the fault detecting method for plural processors mutually connected by bus.

CONSTITUTION: The transmission order of operation confirming signals is set concerning respective plural processors 10-1 to 10-n, and each processor 10-k is provided with a processor state managing means 11 for storing the operation confirm signal transmission orders and operating states of all the processors. When the operation confirm signal is received from a processor 10-(k-1) in the preceding order, after a response signal is returned, the operation confirming signal is transmitted to a processor 10-(k+1) in the next order. When no response signal is received within prescribed time, the fault of the transmission destination processor is reported to all the other processors and the respective processors, which receive the report, change the processor of the processor state managing means 11 into a non-operating state. Then, the processor turning the processor in the next order into the non-operating state transmits the operation confirm signal to the processor in the operating state after the next order.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

BEST AVAILABLE COPY

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平7-262042

(43)公開日 平成7年(1995)10月13日

(51)Int.Cl.⁶

G 0 6 F 11/22

識別記号

3 6 0 J

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数2 O L (全 18 頁)

(21)出願番号 特願平6-46245

(22)出願日 平成6年(1994)3月17日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72)発明者 城 順子

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(74)代理人 弁理士 井桁 貞一

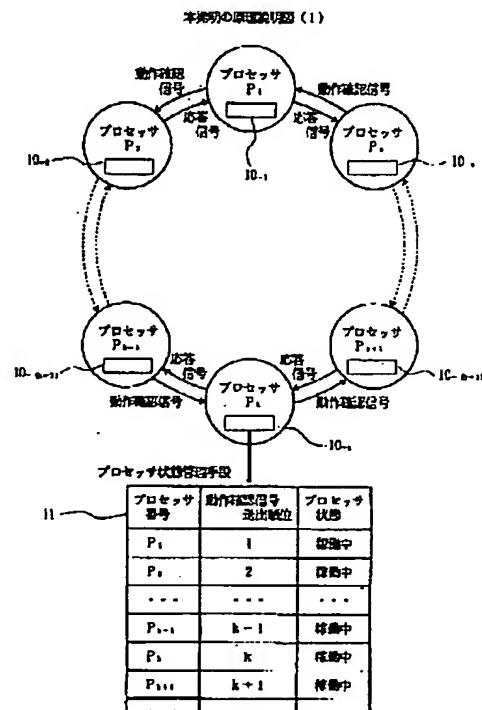
(54)【発明の名称】 プロセッサ障害検出方法

(57)【要約】

【目的】 バスによって互いに接続された複数のプロセッサの障害検出方法に関し、プロセッサの処理負担が少ない障害検出方法を提供することを目的とする。

【構成】 複数のプロセッサ10₁～10_nの各々について動作確認信号の送出順位を設定し、各プロセッサ10

は、全プロセッサの動作確認信号送出順位と稼働状態を記憶するプロセッサ状態管理手段11を備え、前順位のプロセッサ10₁～10_nより動作確認信号を受信したときに応答信号を返送したのち、次順位のプロセッサ10₁～10_nに動作確認信号を送信し、所定時間内に応答信号を受信しないときは他の全プロセッサに送信先プロセッサの障害を通知し、通知を受信した各プロセッサはプロセッサ状態管理手段の当該プロセッサを非稼働状態に変更し、次順位のプロセッサが非稼働状態となったプロセッサは次順位以後の稼働状態にあるプロセッサに動作確認信号を送出するように構成する。



BEST AVAILABLE COPY

【特許請求の範囲】

【請求項 1】 バスによって互いに接続された複数のプロセッサ (10₁ ~ 10_n) の各々が他のプロセッサに動作確認信号を送出してプロセッサの障害検出を行うプロセッサ障害検出方法であって、

前記複数のプロセッサ (10₁ ~ 10_n) の各々について前記動作確認信号を送出する順位を定め、前順位のプロセッサ (10₁) より動作確認信号を受信したプロセッサ (10₁) が次順位のプロセッサ (10₂) に動作確認信号を送出することにより動作確認信号が全プロセッサ (10₁ ~ 10_n) 間を順次リング状に送信されるように構成し、

前記複数のプロセッサ (10₁ ~ 10_n) の各プロセッサ

(10_i) は、前記全プロセッサ (10₁ ~ 10_n) の動作確認信号送出順位と稼働状態を記憶するプロセッサ状態管理手段(11)を備え、かつ、

前順位のプロセッサ (10₁) より動作確認信号を受信したときに正常な状態にあれば該前順位のプロセッサ

(10₁) に応答信号を返送したのち、次順位のプロセッサ (10₂) に動作確認信号を送信して送信先プロセッサ (10₂) より応答信号の有無を監視し、前記送信先プロセッサ (10₂) より所定の時間内に応答信号を受信したときは前記の監視を停止し、所定の時間内に応答信号を受信しなかったときは該送出先プロセッサ (10₂) が障害であると判定して自プロセッサ (10₁) 内の前記プロセッサ状態管理手段(11)に記憶されている当該プロセッサ (10₂) の稼働状態を非稼働状態に変更するとともに、障害と判定したプロセッサ (10₂) を除く他の全プロセッサ (10₁ ~ 10_n) に対して障害と判定したプロセッサ (10₂) の識別情報を付して障害通知を行い、

該障害通知を受信した各プロセッサ (10₁ ~ 10_n) は、各々、自プロセッサ内の前記プロセッサ状態管理手段(11)に記憶されている当該プロセッサ (10₂) の稼働状態を非稼働状態に変更し、次順位のプロセッサ (10₃) が非稼働状態となったプロセッサ (10₁) は、動作確認信号を送出する状態となったとき、次順位以後において稼働状態にある最初の順位のプロセッサ (10₃) に対して動作確認信号を送出することを特徴とするプロセッサ障害検出方法。

【請求項 2】 バスによって互いに接続された複数のプロセッサ (20₁ ~ 20_n) 間に動作確認情報(22)を巡回させてプロセッサ障害の検出を行うプロセッサ障害検出方法であって、

前記動作確認情報(22)内に、該動作確認情報(22)が前記全プロセッサ (20₁ ~ 20_n) 間を一巡するよう、各プロセッサ (20_i) が受信した動作確認情報(22)の送信先プロセッサ (20₁) を指定し、

(20₁) は、前記全プロセッサ (20₁ ~ 20_n) の動作確認情報送出順位と稼働状態を記憶するプロセッサ状態管理手段(21)を備え、かつ、

前順位のプロセッサ (20₁) より動作確認情報(22)を受信したときに正常な状態にあれば該動作確認情報(22)に指定されている送信先プロセッサ (20₁) に該動作確認情報(22)を送信するとともに、該動作確認情報(22)が他の全プロセッサ (20₂ ~ 20_n) を一巡して自プロセッサ (20₁) に戻るまでの時間を監視し、前記動作確認情報(22)を送出してから所定の時間内に該動作確認情報(22)を再び受信したときは前記監視を停止し、所定の時間内に前記動作確認情報(22)を受信しなかったときは他の全プロセッサ (20₂ ~ 20_n) に対して障害プロセッサ確認のための情報であることを識別する情報を付した第 2 の動作確認情報を送信して送信先のプロセッサ (20₂ ~ 20_n) が正常な状態にあるときに返送される応答を待ち、

前記第 2 の動作確認情報を受信した各プロセッサ (20₂ ~ 20_n) は正常な状態にあれば該第 2 の動作確認情報を送信先に指定された第 2 の動作確認情報の送信元プロセッサ (20₁) に対して返送し、前記第 2 の動作確認情報の送信元プロセッサ (20₁)

は、該第 2 の動作確認情報を返送しないプロセッサ (20₂) があつたときに該プロセッサ (20₂) が障害であると判定して自プロセッサ (20₁) 内の前記プロセッサ状態管理手段(21)に記憶されている当該プロセッサ (20₂) の稼働状態を非稼働状態に変更するとともに、障害と判定したプロセッサ (20₂) を除く他の全プロセッサ (20₃ ~ 20_n) に対して障害と判定したプロセッサ (20₂) の識別情報を付して障害通知を行い、

該障害通知を受信した各プロセッサ (20₃ ~ 20_n) は、各々、自プロセッサ内の前記プロセッサ状態管理手段(21)に記憶されている当該プロセッサ (20₂) の稼働状態を非稼働状態に変更し、前記障害通知を受信したときに巡回中の動作確認情報(22)を受信しているプロセッサは、該動作確認情報(22)中で送信先プロセッサとして指定されている前記障害と判定されたプロセッサ (20₂) を該プロセッサの次順位のプロセッサ (20₃) に変更したのち、該動作確認情報(22)を指定された送信先プロセッサに送信することを特徴とするプロセッサ障害検出方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、バスによって互いに接続された複数のプロセッサの障害検出方法に関する。

【0002】 近年、バスに接続された複数の処理プロセッサが分担して処理を行うシステムが増加しているが、

管理や共通制御を行う上位のプロセッサによって行われていた共通的な処理や処理プロセッサの障害監視などを複数の処理プロセッサが行うようになってきている。

【0003】このように処理プロセッサの障害を複数の処理プロセッサ自身で検出する方法として、従来技術では各プロセッサが他の全プロセッサの障害の有無を定期的に確認する方法がとられている。しかし、この方法はプロセッサの台数が増えるに従って障害監視のためのプロセッサの処理負担が急激に増加すると言う性質を有している。

【0004】このため、障害検出のためのプロセッサの処理負担が小さいプロセッサ障害検出方法が求められている。

【0005】

【従来の技術】図21はプロセッサ障害検出の対象となるシステムの構成図、図22は従来技術のプロセッサ障害検出方法の説明図である。

【0006】図21の(1)はプロセッサが階層構造となっているシステムの構成例を示している。図のシステムでは複数の回線処理プロセッサ31が図示省略された回線の信号処理を行っているが、回線処理プロセッサ31は複数台ごとにグループ化され、各グループごとに設けられた通信制御プロセッサ32が共通的な通信制御の処理を行っている。また、複数の通信制御プロセッサ32の上位にはシステム全体を管理する管理プロセッサ33が設けられ、システム全体の管理を行っている。

【0007】このような構成では回線処理プロセッサ31の障害はその上位にある通信制御プロセッサ32によって監視され、障害の発生が検出されると通信制御プロセッサ32より管理プロセッサ33に報告され、管理プロセッサ33から他の通信制御プロセッサ32、或いは必要に応じて各回線処理プロセッサ31に障害の発生が通知される。

【0008】しかし、プロセッサの小型化と高性能化の著しい進展に伴い、最近ではこれまで通信制御プロセッサや管理プロセッサによって行われていたシステムの共通的な処理が回線処理プロセッサに移されるようになってきている。

【0009】図21の(2)はこのような上位プロセッサを持たないシステムの構成の一例を示している。図示のように、このシステムは処理を対等に分担して行う複数の回線処理プロセッサ34のみで構成されており、各回線処理プロセッサ34は図21の(1)の通信制御プロセッサ22及び管理プロセッサ23の役割をも備えたものとなっている。

【0010】図21の(2)に示した回線処理プロセッサ34のようなプロセッサ(以下、単にプロセッサと記す)からなるシステムにおいてプロセッサの障害を検出する方法として、各プロセッサ34がそれぞれ他の全プロセッサ34の障害の有無を確認する方法が従来から行われてい

に対して例えばヘルスチェック信号と呼ばれる障害の有無を確認する信号を定期的に送信し、ヘルスチェック信号を受信したプロセッサ34は正常な状態にあれば送信元のプロセッサ34に応答信号を返送する。応答信号がない場合にはそのプロセッサ34を障害と判定する。

【0011】図22は以上のような従来技術のプロセッサ障害検出方法におけるヘルスチェック信号の送受信関係を説明する図である。同図に示すように従来技術のプロセッサ障害検出方法は、メッシュ状の相互監視型障害検出方法になっているため、プロセッサ34の台数が増加すると各プロセッサ34間に送受信されるヘルスチェック信号や応答信号の量は急激に増加する。このため、各プロセッサ34は障害検出のための処理負担が増加し、本来行うことになっている処理の能力が低下し、システム全体の処理能力も低下する。

【0012】

【発明が解決しようとする課題】以上のように、バスによって互いに接続された複数のプロセッサからなるシステムにおける従来技術のプロセッサ障害検出方法は、各プロセッサがメッシュ状に障害検出のための信号を送受信するため、プロセッサが増加すると各プロセッサの障害検出のための処理負担が急激に増加し、システム全体の処理能力が低下すると言う問題を有している。

【0013】本発明は、プロセッサの処理負担が小さい障害検出方法を提供することを目的とする。

【0014】

【課題を解決するための手段】図1及び図2は本発明の原理説明図である。図中、10₁～10_n及び20₁～20_nはバス(図示省略)によって互いに接続された複数のプロセッサ、11は複数のプロセッサ10₁～10_nの各々に備えられ、全プロセッサ10₁～10_nの動作確認信号送出順位と稼働状態を記憶するプロセッサ状態管理手段、21は複数のプロセッサ20₁～20_nの各々に備えられ、全プロセッサ20₁～20_nの動作確認情報送出順位と稼働状態を記憶するプロセッサ状態管理手段である。また、22はプロセッサ20₁～20_nの障害を検出するために全プロセッサ20₁～20_n間に巡回される動作確認情報である。

【0015】図1は、バスによって互いに接続された複数のプロセッサ10₁～10_nの各々が他のプロセッサに動作確認信号を送出してプロセッサ障害の検出を行うプロセッサ障害検出方法の原理を説明する図である。

【0016】図1においては、複数のプロセッサ10₁～10_nの各々、例えばプロセッサ10₁(以下、プロセッサ10₁を個々のプロセッサを代表するプロセッサとして説明する)について動作確認信号を送出する順位を定め、前順位のプロセッサ10₁より動作確認信号を受信したプロセッサ10₁が次順位のプロセッサ10₂に動作確認信号を送出することにより動作確認信号が全プロセッサ10₁～10_n間を順次リング状に送信されるように構

【0017】各プロセッサ10_iは、前順位のプロセッサ10_(i-1)より動作確認信号を受信したときに正常な状態にあれば前順位のプロセッサ10_(i-1)に応答信号を返送したのち、次順位のプロセッサ10_(i+1)に動作確認信号を送信して送信先のプロセッサ10_(i+1)よりの応答信号の有無を監視する。

【0018】プロセッサ10_iは動作確認信号の送信先プロセッサ10_(i+1)より所定の時間内に応答信号を受信したときは応答信号の監視を停止し、所定の時間内に応答信号を受信しなかったときは動作確認信号送信先のプロセッサ10_(i+1)が障害であると判定して自プロセッサ10_i内のプロセッサ状態管理手段11に記憶されている当該プロセッサ10_(i+1)の稼働状態を非稼働状態に変更するとともに、障害と判定したプロセッサ10_(i+1)を除く他の全プロセッサ10₁～10_(i-1)、10_(i+2)～10_nに対して障害プロセッサ10_(i+1)の識別情報を付して障害を検出したことを知らせる通知（以下、障害通知と記す）を行う。

【0019】この障害通知を受信した各プロセッサ10₁～10_(i-1)、10_(i+2)～10_nは、各々、自プロセッサ内のプロセッサ状態管理手段11に記憶されている当該プロセッサ10_(i+1)の稼働状態を非稼働状態に変更する。

【0020】非稼働状態のプロセッサが発生したのち、次順位のプロセッサ10_(i+1)が非稼働状態となったプロセッサ10_iは、動作確認信号を送出する状態となったとき、次順位以後において稼働状態にある最初の順位のプロセッサ10_(i+2)に対して動作確認信号を送出する。

【0021】図2はバスによって互いに接続された複数のプロセッサ20₁～20_n間に動作確認情報22を巡回させてプロセッサ障害の検出を行うプロセッサ障害検出方法の原理を説明する図である。

【0022】図2においては、動作確認情報22内に、動作確認情報22が全プロセッサ20₁～20_n間を一巡するよう、各プロセッサ20_iが受信した動作確認情報22の送信先プロセッサ20_(i+1)を指定しておく。

【0023】複数のプロセッサ20₁～20_nの各々、例えばプロセッサ20₁（以下、プロセッサ20_iを個々のプロセッサを代表するプロセッサとして説明する）は、前順位のプロセッサ20_(i-1)より動作確認情報22を受信したときに正常な状態にあればその動作確認情報22に指定されている送信先プロセッサ20_(i+1)にその動作確認情報22を送信するとともに、その動作確認情報22が他の全プロセッサ20_(i+2)～20_(i+1)を一巡して自プロセッサ20_iに戻るまでの時間を監視する。

【0024】プロセッサ20_iは、動作確認情報22を送出してから所定の時間内にその動作確認情報22を再び受信したときは時間の監視を停止し、所定の時間内に動作確認情報22を受信しなかったときは他の全プロセッサ20₁～20_(i-1)、20_(i+1)～20_nに対して障害プロセッサ確

の動作確認情報（図示省略）22認用情報を送信して送信先のプロセッサが正常な状態にあるときに返送される応答を待つ。

【0025】第2の動作確認情報を受信した各プロセッサ20₁～20_(i-1)、20_(i+1)～20_nは、正常な状態にあればその第2の動作確認情報中に指定された送信先プロセッサ、即ち、第2の動作確認情報の送信元プロセッサ20_iに対してその第2の動作確認情報を返送（送信）する。

【0026】第2の動作確認情報の送信元プロセッサ20_iは、その第2の動作確認情報を返送しないプロセッサ（プロセッサ20_(i+1)とする）があると、そのプロセッサ20_(i+1)が障害であると判定して自プロセッサ20_i内のプロセッサ状態管理手段21に記憶されている当該プロセッサ20_(i+1)の稼働状態を非稼働状態に変更するとともに、障害と判定したプロセッサ20_(i+1)を除く他の全プロセッサ20₁～20_(i-1)、20_(i+2)～20_nに対して障害と判定したプロセッサ20_(i+1)の識別情報を付して障害通知を行う。

【0027】障害通知を受信した各プロセッサ20₁～20_(i-1)、20_(i+2)～20_nは、各々、自プロセッサ内のプロセッサ状態管理手段21に記憶されている当該プロセッサ20_(i+1)の稼働状態を非稼働状態に変更する。

【0028】障害通知を受信したときに巡回中の動作確認情報22を受信しているプロセッサは、その動作確認情報22中で送信先プロセッサとして指定されている障害と判定されたプロセッサ20_(i+1)をそのプロセッサの次順位のプロセッサ20_(i+2)に変更したのち、その動作確認情報22を指定された送信先プロセッサに送信する。

【0029】以上のように、図1においては、各プロセッサ10_iはプロセッサ障害の有無を確認するための動作確認信号を前順位のプロセッサ10_(i-1)より受信したときに正常状態にあれば応答信号を返し、次いで次順位のプロセッサ10_(i+1)に動作確認信号を送信し、送信先のプロセッサ10_(i+1)より応答があるか否かによって送信先プロセッサ10_(i+1)の障害検出を行う。

【0030】また、図2においては、各プロセッサ20_iはプロセッサの監視を行う場合に前順位のプロセッサ20_(i-1)より動作確認情報22を受信したときに正常状態にあればその動作確認情報22を次順位のプロセッサ20_(i+1)に送出し、送出してからその動作確認情報22が他の全プロセッサ20_(i+2)～20_(i+1)を一巡して自プロセッサ20_iに戻るまでの時間を監視することによって、他のプロセッサ20_(i+2)～20_(i+1)の障害の有無を検出する。障害が検出された場合、障害を検出したプロセッサは第2の動作確認情報を各プロセッサに送信してその返送の有無によって障害プロセッサを特定する。

【0031】図1及び図2の何れの方法も各プロセッサ10_iは定められた前順位のプロセッサ10_(i-1)より動作

られた後位のプロセッサ10_iのみに対して動作確認信号または動作確認情報22を送信するようになっている。

【0032】

【作用】図1のプロセッサ障害検出方法においては、各プロセッサ10_iは前順位のプロセッサ10_{i-1}より動作確認信号を受信したときに正常状態にあれば応答信号を返したのち、次順位のプロセッサ10_{i+1}に動作確認信号を送信し、送信先のプロセッサ10_{i+1}より応答があるか否かによって送信先プロセッサ10_{i+1}の障害検出を行う。即ち、各プロセッサは複数のプロセッサのうち、予め定められた前順位と次順位のプロセッサとの間で動作確認信号の送受信と応答信号の送受信を行うのみでプロセッサ障害を検出するので、プロセッサの数が多くなっても各プロセッサにおける障害検出のための処理が増加することがない。

【0033】また、図2のプロセッサ障害検出方法においては、各プロセッサ20_jは前位のプロセッサ20_{j-1}より動作確認情報22を受信したときにその動作確認情報22を次順位のプロセッサ20_{j+1}に送信し、送信してからその動作確認情報22が他の全プロセッサ20_k（ $k = j+2, \dots, N$ ）を一巡して自プロセッサ20_jに戻るまでの時間を監視し、所定の時間内に動作確認情報22が返送されてこなかった場合に他のプロセッサ20_k（ $k = j+2, \dots, N$ ）のいずれかに障害があると判断するため、図1と同様、プロセッサの数が多くなっても各プロセッサがプロセッサ障害の有無を監視するための処理が増加することがない。

【0034】また、障害があると判断したときは、その判断を行ったプロセッサから全プロセッサに確認のための第2の動作確認情報を送って応答のないプロセッサを障害プロセッサと判定するが、この処理は複数のプロセッサのいずれかに障害が発生したときのみ、複数のプロセッサのひとつによって行われるものであるため、その処理がシステム全体の処理能力に影響を及ぼすことは殆どない。

【0035】図1及び図2から明らかなように、図1及び図2の方法はリング型障害検出方法であり、従来技術において説明したメッシュ状の相互監視型障害検出方法と異なり、各プロセッサが障害検出を行う相手のプロセッサはそれぞれ1台に限定されるため、プロセッサの台数が増加しても各プロセッサの障害検出のための処理負担が増加することがなく、システム全体の処理能力が低下することがない。

【0036】

【実施例】図3は本発明の実施例プロセッサ状態管理テーブル構成説明図、図4乃至図11は本発明の図1の原理（以下、応答監視型と記す）に基づくプロセッサ障害検出方法の実施例を示し、図4乃至図6は本発明の実施例情報送受信経路説明図（応答監視型）、図7乃至図9は

型）、図10及び図11は本発明の実施例動作フロー図（応答監視型）である。

【0037】図12乃至図20は本発明の図2の原理（以下、巡回型と記す）に基づくプロセッサ障害検出方法の実施例を示し、図12及び図13は本発明の実施例ヘルスチェックリスト構成説明図、図14乃至図17は本発明の実施例情報送受信経路説明図（巡回型）、図18乃至図20は本発明の実施例情報送受信シーケンス図（巡回型）である。

【0038】全図を通じ、同一符号は同一対象物を示し、10_i～10_j及び20_i～20_jはプロセッサ、11及び21はプロセッサ状態管理テーブル、22はヘルスチェックリストである。なお、プロセッサ状態管理テーブル11、12はそれぞれ図1及び図2におけるプロセッサ状態記憶手段11、12の実現形態、ヘルスチェックリスト22は図2における動作確認情報22の実現形態である。

【0039】各図におけるプロセッサ10_i～10_j及びプロセッサ20_i～20_jは例えば図21の(2)の回線処理プロセッサ34のように、互いにバス（図示省略）によって接続された複数のプロセッサである。上記各図にはそれぞれ6台のプロセッサを図示しているが、以下、説明の便からプロセッサ10_i～10_j及びプロセッサ20_i～20_jに同一のプロセッサ番号P_i～P_jを付与し、プロセッサ10_i～10_j及びプロセッサ20_i～20_jをプロセッサP_i～P_jと記す。

【0040】最初に図3乃至図16により本発明における応答監視型のプロセッサ障害検出方法の実施例を説明する。応答監視型のプロセッサ障害検出方法では各プロセッサP_i～P_jは他のプロセッサの障害を検出するために動作確認信号（以下、ヘルスチェック信号と記す）を送出し、そのヘルスチェック信号に対して相手プロセッサから応答があるか否かによって相手プロセッサの障害の有無を判定するが、本発明による応答監視型のプロセッサ障害検出方法ではヘルスチェック信号を受信する相手のプロセッサと、ヘルスチェック信号を送信する相手のプロセッサはそれぞれ1台に限られ、全プロセッサP_i～P_jがリング状に順次ヘルスチェック信号の送受信を行うようになっている。

【0041】このため、本発明では全プロセッサP_i～P_jのヘルスチェック信号の送出順序を予め定めて各プロセッサP_i～P_jに備えられるプロセッサ状態管理テーブル11に記憶させ、各プロセッサP_i～P_jは前順位のプロセッサよりヘルスチェック信号を受信するとそのプロセッサに対して応答信号を返送したのち、次順位のプロセッサにヘルスチェック信号を送出するようにしている。

【0042】図3にはプロセッサ状態管理テーブル11の記憶内容の一例が図示されている。図3に示されているように、プロセッサ状態管理テーブル11には全プロセッサ

。のヘルスチェック信号の送出順位が指定されている。図3ではプロセッサP₁からプロセッサP_nまで番号順にヘルスチェック信号の送信が行われる例を示している。また、プロセッサ状態管理テーブル11のプロセッサ状態の欄には各プロセッサP₁～P_nが正常に動作しているか否かが記憶されているが、図3の(1)では全プロセッサP₁～P_nがすべて稼働状態にある例が示されている。

【0043】図4はプロセッサ状態管理テーブル11の記憶内容が図3の(1)である場合、即ち、全プロセッサP₁～P_nが稼働中であり、ヘルスチェック信号の送出順序がプロセッサの番号順となっている場合におけるヘルスチェック信号と応答信号が送受信される経路を示している。この例ではプロセッサP₁が最初にヘルスチェック信号を送出するが、その送出時期はタイマなど公知の技術を用いて設定するものとする。

【0044】いま、プロセッサP₁よりヘルスチェック信号がプロセッサP₂に対して送出されたものとする。プロセッサP₂は予め定められているヘルスチェック信号を受信したときに行う処理を実行し、正常に処理を終わるとプロセッサP₁に対して応答信号を返送する。なお、ヘルスチェック信号及び応答信号は図示省略されたバスを介して送受信される。

【0045】次いでプロセッサP₂は自プロセッサ内に記憶しているプロセッサ状態管理テーブル11に指定されている次順位のプロセッサP₃に対してヘルスチェック信号を送信する(次順位のプロセッサの選別方法は後述)。プロセッサP₃も正常であれば応答信号が返送され、以下、障害がなければ各プロセッサP₁～P_nにおいても同様の処理が行われる。これにより、全プロセッサP₁～P_nに障害がないことが確認される。

【0046】図7はヘルスチェック信号と応答信号の送受信シーケンスを示している。ヘルスチェック信号の送出順位が最初となっているプロセッサP₁は、最後のプロセッサP_nからヘルスチェック信号を受信して応答信号を返送したのち、引き続いて次順位のプロセッサP₂に対してヘルスチェック信号を送出するか、タイマ(図示省略)などにより時間を置いてから改めて次のヘルスチェック信号を送出するかはシステムによって任意に設定できるが、図7には引き続いてヘルスチェック信号の送出を行う例を図示している。

【0047】次に、プロセッサP₁～P_nの何れか1つ、例えばプロセッサP₁に障害が発生した場合について説明する。図5はその状態を図示したものであるが、この場合は図示のように、プロセッサP₁がプロセッサP₂に対してヘルスチェック信号を送出したときにプロセッサP₂より応答信号が返送されない。前記したように、各プロセッサP₁～P_nはヘルスチェック信号を送出したときに応答監視用のタイマ(図示省略)を起動

ッサより応答信号を受信したときは相手プロセッサが正常であるとしてタイマを停止し、応答信号が返送されずにタイムアウトとなったときは相手プロセッサが障害であると判定する。

【0048】プロセッサP₂はプロセッサP₁より応答がないことによりプロセッサP₁が障害であると判定すると、自プロセッサP₂内のプロセッサ状態管理テーブル11のプロセッサP₁の状態を「稼働中」から「非稼働」に変更するとともに、稼働中の他の全プロセッサP₁～P_n、P₃～P_nに対してプロセッサP₁が障害であることを通知する(図5に点線で示す)。

【0049】障害通知を受けた各プロセッサP₁～P_n、P₃～P_nはそれぞれ自プロセッサ内のプロセッサ状態管理テーブル11のプロセッサP₁の状態を非稼働に変更する。これによって、プロセッサP₁～P_n、P₃～P_nのプロセッサ状態管理テーブル11のプロセッサP₁の状態はすべて非稼働に変更されるが、図3の(2)はその状態を示している。

【0050】上記の障害通知の処理が終了すると再びヘルスチェック信号の送信が開始されるが、次順位のプロセッサが非稼働状態となっているプロセッサはヘルスチェック信号をその次のプロセッサに送信するようになっている。上記の例ではプロセッサP₂は次順位のプロセッサP₃が非稼働状態になっている間はその次の順位のプロセッサP₄に対してヘルスチェック信号を送出する(詳細は後述)。

【0051】図8は以上の情報送受信シーケンスを示している。図はプロセッサP₂内に設定される応答監視用のタイマの時限がt₁秒であり、プロセッサP₂がプロセッサP₁の障害通知の終了に引き続いてプロセッサP₃に対してヘルスチェック信号を送信する例を示している。なお、図8における「障害通知(P₁)」はプロセッサP₂の障害を検出したことを通知する障害通知であることを示しているが、各プロセッサP₁～P_n、P₃～P_nに対する障害通知の順序は一例を示すものであり、図示のものに限られない。

【0052】次に、プロセッサP₁の障害が回復し、再びシステムに組み込まれる場合について説明する。図6はプロセッサP₂を組み込む前の信号送受信状態を示しているが、図中に実線で示したように、プロセッサP₂はプロセッサP₁に対してヘルスチェック信号を送出している。この状態でプロセッサP₁の障害が回復し、システムに組み込める状態になると、プロセッサP₁より各プロセッサP₁～P_n、P₃～P_nに対して組み込み通知を行う(図6に点線で示す)。

【0053】組み込み通知を受けた各プロセッサP₁～P_n、P₃～P_nはそれぞれ自プロセッサ内のプロセッサ状態管理テーブル11のプロセッサP₁の状態を「非稼働」から「稼働中」に変更する。これによって、全プロ

ロセッサP_kの状態はすべて非稼働に変更され、図3の(2)の状態から同図(1)の状態に戻り、以後、ヘルスチェック信号の送受信は図4及び図7におけると同様、プロセッサP_kを含めて行われる。

【0054】図9は以上の組み込み前後の信号送受信シーケンスを示しているが、図ではプロセッサP_kがプロセッサP_{k+1}との間でヘルスチェック信号の送信と応答が行われた直後に組み込み通知が行われた例を示している。この場合は、組み込み通知の処理が終わるとプロセッサP_kよりプロセッサP_{k+1}に対するヘルスチェック信号の送信からヘルスチェックが再開される。

【0055】次に各プロセッサP₁～P₆のヘルスチェック動作の詳細について図10及び図11により説明する。図10はヘルスチェック信号を送出する場合の動作フロー、図11はヘルスチェック信号を受信した場合の動作フローを示しているが、説明の便から、図11から説明する。なお、図10及び図11は何れもプロセッサP_k (k=1～6)における処理の例を示すものとする。また、以下における括弧内のS1～S26は図10または図11内の関連するステップの符号である。

【0056】図11においてプロセッサP_kは前順位のプロセッサP_{k-1}からヘルスチェック信号を受ける(S21)と、所定のヘルスチェック処理を実行する(S22)。処理結果が良好であればプロセッサP_{k-1}に対して応答信号を返送し、図10に示すヘルスチェック信号送信処理に移る(S23→S24→S25)。所定のヘルスチェック処理を実行したときに障害が存在すれば良好な結果が得られないため応答信号を返送できずに処理を終わる(S23→S26)。

【0057】次に、プロセッサP_kがヘルスチェック信号を送信する場合の動作を図10により説明する。ヘルスチェック信号を送信する状態になると、プロセッサP_kは図示省略されたカウンタの数値xに“0”を設定(初期化)し(S1)、次いでカウンタの数値に“1”を加える(S2)。

【0058】次に、プロセッサ状態管理テーブル11にアクセスし、自プロセッサP_kの番号である“k”に前記の“x”を加えた“k+x”をインデックスとしてプロセッサ状態管理テーブル11に記憶されている情報(以下、データと記す)を読み取る(S3)。なお、この状態の図10のS2ではx=1であるため、使用するインデックス“k+x”は“k+1”となるが、以下、図3を用いてプロセッサ状態管理テーブル11の読み取り処理について説明する。

【0059】図3に示すようにプロセッサ状態管理テーブル11にはインデックスが付してあるが、図3の(1)ではヘルスチェック信号送出順位がプロセッサの番号順になっているため、プロセッサP_kのプロセッサ状態管理テーブル11のインデックス“k+1”は自プロセッサP

プロセッサP_kをプロセッサP_{k+1}とすると、k=3となり、インデックスは(k+1)=4となるため、プロセッサP_kはプロセッサ状態管理テーブル11のインデックス4にアクセスしてそのデータを読むことになる。ここで読み出されるプロセッサがヘルスチェック信号の送信先となるが、そのプロセッサは自プロセッサP_kの次順位のプロセッサP_{k+1}であることが確認され、かつ、そのプロセッサP_{k+1}は稼働中であることも判る。

【0060】以下、図10に戻って説明する。以上により次順位のプロセッサがプロセッサP_{k+1} (この場合はP₄)であり、稼働中であることが確認されると、プロセッサP_kは応答監視タイマ(図示省略)に例えばt₁秒を設定(S4→S5)したのち、プロセッサP_{k+1}にヘルスチェック信号を送出し(S6)、同時に応答監視タイマの計数を開始する(S7)。

【0061】t₁秒が経過しないうち、即ち、応答監視タイマがタイムアウトにならないうちにプロセッサP_{k+1}より応答信号を受信した場合は次順位のプロセッサP_{k+2}は正常であると判定し、応答監視タイマの計数を停止してヘルスチェック信号の送信処理を終わる(S8, S9)。これにより、次順位プロセッサの障害検出処理が終わることになる。もし、t₁秒が経過しても応答信号を受信しない場合は、応答監視タイマがタイムアウトとなるので、プロセッサP_kは次順位のプロセッサP_{k+1}が異常であると判定し、自プロセッサ内のプロセッサ状態管理テーブル11のプロセッサP_{k+1} (この場合はP₄)の状態を「非稼働」に変更する(S10)。次いで、プロセッサP_kはプロセッサ状態管理テーブル11により稼働状態にあるプロセッサを確認し、稼働中の全プロセッサに対してプロセッサP_{k+1}が非稼働状態にあることを通知する(S11)。

【0062】以上の障害通知を終わると、プロセッサP_kはカウンタの数値xに“1”を加え(S2)、前と同じ送信処理を行う。これにより、プロセッサP_kはプロセッサP_{k+1}の次の順位にあるプロセッサP_{k+2}に対してヘルスチェック信号の送出を行うことになる。

【0063】次に、図3と図12乃至図20により本発明における巡回型のプロセッサ障害検出方法の実施例を説明する。巡回型のプロセッサ障害検出方法では各プロセッサP₁～P₆は他のプロセッサより動作確認情報22(以下、ヘルスチェックリストと記す)を受信するとそのヘルスチェックリスト22に指定されているプロセッサに送出し、送出後一定時間以内にそのヘルスチェックリスト22が全プロセッサP₁～P₆を巡回して再び自プロセッサに戻ってきたか否かにより他プロセッサの中に異常なプロセッサがあるか否かを判定する。

【0064】上記のヘルスチェックリスト22はそのヘルスチェックリスト22を送信するプロセッサにより作成されるが、図12及び図13にヘルスチェックリスト22の構成

あり、ひとは図12の(1)～(3)に示す通常モードのヘルスチェックリストで、障害の有無を確認するために巡回させるヘルスチェックリストである。他は図13の(1)に示す障害モードのヘルスチェックリストで、障害プロセッサが存在することが確認された場合に、障害プロセッサを特定させるために障害の存在を検出したプロセッサから他の全プロセッサに対して送信されるものである。

【0065】図12及び図13に示すように、通常モードのヘルスチェックリスト22はモードの欄に通常モードであることを示す情報が記憶され、障害モードのヘルスチェックリスト22はモードの欄に障害モードであることを示す情報が記憶される。なお、以上における障害モードのヘルスチェックリスト22は図2の説明において第2の動作確認情報と記載されたものに相当する。図2の説明では図2に図示された動作確認情報22と区別するため第2の動作確認情報については符号を付していなかったが、以下においては障害モードのヘルスチェックリストについても通常モードのヘルスチェックリスト22と同一の符号22を使用する。

【0066】通常モード及び障害モードのヘルスチェックリスト22の何れにも、受信プロセッサ番号、即ち、ヘルスチェックリスト22を受信したプロセッサの番号と、送信先プロセッサ番号、即ち、ヘルスチェックリスト22を受信したプロセッサがそのヘルスチェックリスト22を送信する相手のプロセッサの番号を記憶する箇所が設けられている。

【0067】以下、巡回型のプロセッサ障害検出方法の動作を上記各図及び図3を用いて説明する。巡回型のプロセッサ障害検出方法においては各プロセッサ内に、全プロセッサ $P_1 \sim P_n$ のヘルスチェックリスト22の送出順位と稼働状態を記憶するプロセッサ状態管理テーブル21が備えられる。図3はプロセッサ状態管理テーブル11とプロセッサ状態管理テーブル21の構成を説明する図であるが、両者は、応答監視型のプロセッサ状態管理テーブル11がヘルスチェック信号の送出順位を設定するのに対して巡回型のプロセッサ状態管理テーブル21ではヘルスチェックリスト22の送出順位を設定する点が異なるのみであり、内容的に大差はないのでプロセッサ状態管理テーブル21の構成については詳細説明を省略する。

【0068】最初に、全プロセッサ $P_1 \sim P_n$ が正常に動作している状態で通常モードのヘルスチェックリスト22を巡回させる場合の動作を説明する。この場合は各プロセッサ $P_1 \sim P_n$ に記憶されているプロセッサ状態管理テーブル21には図3の(1)に図示されているように全プロセッサ $P_1 \sim P_n$ が稼働中であることが記憶され、同時にヘルスチェックリスト22がプロセッサ P_1 より番号順に送出されることが示されている。

【0069】通常モードのヘルスチェックリスト22の送

も可能であるが、プロセッサ状態管理テーブル11において送出順位が1番となっているプロセッサ P_1 から行うのが最も一般的であるので、プロセッサ P_1 がヘルスチェックリスト22を作成する例について説明する。なお、ヘルスチェックリスト22は他のプロセッサ間を転送されるときはデータ・フレームの形式となっているため、ヘルスチェックリスト・フレームと呼ぶのが適当であるが、以下においてはプロセッサ内に記憶されている状態のヘルスチェックリストも、データとして転送されている状態のヘルスチェックリストも区別することなく、ヘルスチェックリストと記す。

【0070】上記においてプロセッサ P_1 が作成するヘルスチェックリスト22は全プロセッサ $P_1 \sim P_n$ に巡回させるヘルスチェックリストであるため、モード欄に「通常モード」を示す所定のコードを設定し、プロセッサ状態管理テーブル21の指定に従ってヘルスチェックリスト22を巡回させるよう、受信プロセッサ番号と送信先プロセッサ番号を図12の(1)に記載されているようにプロセッサ番号を記憶させる。

【0071】ヘルスチェックリスト22を作成するとプロセッサ P_1 は、自プロセッサに関するデータが記憶されているインデックス“1”（インデックスの数字は受信プロセッサ番号と一致するものとする）のデータを読み出して送信先プロセッサの番号が P_2 であることを確認するとともに、プロセッサ状態管理テーブル21によってプロセッサ P_2 が稼働中であるか否かを確認する。

【0072】プロセッサ P_1 はプロセッサ P_2 が稼働中であることを確認すると、作成したヘルスチェックリスト22をプロセッサ P_2 に送信する。また、このとき巡回時間監視タイマ（図示省略）に時間を設定して起動させる。この場合に設定される時間は、異常がない場合にヘルスチェックリスト22が全プロセッサ $P_1 \sim P_n$ を一巡して自プロセッサ P_1 に戻るまでの時間を或る程度の余裕をもってカバーする時間とする。なお、ヘルスチェックリスト22も図示省略されたバスを介して送信される。

【0073】プロセッサ P_2 はプロセッサ P_1 からヘルスチェックリスト22を受信すると、受信したヘルスチェックリスト22の受信プロセッサ番号を検索し、自プロセッサ P_2 の番号が記載されているデータの内容を読み取る。この場合は図12の(1)のインデックス“2”のデータが読み取られるが、受信したヘルスチェックリスト22が通常モードのものであり、その送信先がプロセッサ P_1 であることが確認されるので、プロセッサ P_2 は受信したヘルスチェックリスト22をプロセッサ P_1 に対して送信する。このとき、プロセッサ P_2 においても巡回時間監視タイマ（図示省略）に時間を設定して起動する。なお、巡回型のプロセッサ障害検出方法ではヘルスチェックリスト22を受信したときに送信元（受信）プロセッサであるプロセッサ P_1 に対して応答信号は返送しな

【0074】以下同様にして、ヘルスチェックリスト22はプロセッサP₁～P_nを巡回し、プロセッサP₁よりプロセッサP₂に対して送信される。プロセッサP₁は巡回時間監視タイマに設定した時間が経過する前にこのヘルスチェックリスト22を受信すると巡回時間監視タイマを止め、巡回時間の監視を解除する。

【0075】プロセッサP₁は受信したヘルスチェックリスト22を前回と同様にして再びプロセッサP₂に対して送信するが、これを受信するとプロセッサP₂も自プロセッサ内の巡回時間監視タイマを停止する。プロセッサP₁、以下についても同様である。

【0076】図14は以上のようにしてヘルスチェックリスト22が巡回する経路を図示したものであり、図18はヘルスチェックリスト22の巡回動作のシーケンスを図示したものであるが、何れも特に説明を要する事項がないので説明は省略する。

【0077】次に、プロセッサP₁～P_nの何れか1つ、例えばプロセッサP₁に障害が発生した場合について説明する。プロセッサP₁に障害が発生した場合は、図14または図18から明らかなように、ヘルスチェックリスト22はプロセッサP₁からプロセッサP₂に送出された段階で巡回が停止するので、ヘルスチェックリスト22はプロセッサP₁～P_nには戻らない。プロセッサP₁～P_nがそれぞれ自プロセッサ内の巡回時間監視タイマに設定した時間が同一であるとする、3つのプロセッサP₁～P_nの中で最も早くヘルスチェックリスト22を送信したプロセッサP₁の巡回時間監視タイマが最初にタイムアウトとなる。

【0078】タイムアウトによりプロセッサP₁はプロセッサ障害の発生を検出するが、この時点ではどのプロセッサが障害であるか識別することはできないので障害プロセッサを特定するために障害モードのヘルスチェックリスト22を稼働中の全プロセッサP₁～P_nに対して送信する。この時点ではプロセッサP₁はまだ稼働中のプロセッサに含まれるので、障害モードのヘルスチェックリスト22は図15に示すようにプロセッサP₂～P_nの全部に送られる。なお、この障害モードのヘルスチェックリスト22も図示省略されたバスを介して送られる。

【0079】図13の(1)は障害モードのヘルスチェックリスト22の構成を示している。この例では障害モードのヘルスチェックリスト22はプロセッサP₁において作成されるが、図示のように、このヘルスチェックリスト22はモード欄に「障害モード」を示す所定のコードを設定し、受信プロセッサ番号には送信先の全プロセッサP₂～P_nの番号を設定し、送信先プロセッサにはすべて自プロセッサの番号であるP₁を設定する。

【0080】この障害モードのヘルスチェックリスト22は前記のようにプロセッサP₁より直接全プロセッサP₂～P_nに送られるが、障害モードのヘルスチェックリ

の場合と同様に送信先プロセッサ番号に指定されたプロセッサにそのヘルスチェックリスト22を送信する。この場合、送信先プロセッサはすべてプロセッサP₁となっているため、返送されるヘルスチェックリスト22は巡回することなく、すべて直接プロセッサP₁に送られる。

【0081】プロセッサP₁は複数のプロセッサから返送される障害モードのヘルスチェックリスト22を受信するが、障害プロセッサであるプロセッサP₁は障害モードのヘルスチェックリスト22を受信したときもそのヘルスチェックリスト22を送信先プロセッサに返送することはないので、プロセッサP₁はヘルスチェックリスト22を返送してこないプロセッサP₁が障害プロセッサであると判定する。

【0082】図15は障害モードのヘルスチェックリスト22が送信される経路を図示し、図19は障害モードのヘルスチェックリスト22の送受信のシーケンスを図示したものである。図15及び図19に示すように、プロセッサP₁のみは障害モードのヘルスチェックリスト22を返送していない。

【0083】プロセッサP₁は障害プロセッサがプロセッサP₁であると判定すると、プロセッサP₁を除く稼働中の全プロセッサP₂～P_nに対してプロセッサP₁が障害であることを通知する。各プロセッサP₂～P_nは障害通知を受信するとプロセッサ状態管理テーブル21のプロセッサP₁の状態を「非稼働」に変更する。図16は障害通知が送信される経路を図示したものであるが、この通知も図示省略されたバスを介して行われる。

【0084】プロセッサP₁は上記の障害通知を行うとともに、自プロセッサ内のプロセッサ状態管理テーブル21のプロセッサP₁の状態を「非稼働」に変更し、更に通常モードのヘルスチェックリスト22を修正する。この修正方法は図12の(2)のA方式と同図(3)のB方式の2種類の方法があるが、システムの条件に従って何れをとってもよい。

【0085】図12の(2)のA方式では図12の(1)におけるインデックス“3”の送信先プロセッサ番号を「P₁」から「P₂」に変更する。この方法ではインデックス“4”にはプロセッサP₁の番号が残っているが、プロセッサP₁よりプロセッサP₁に対してヘルスチェックリスト22が送られなくなるので、プロセッサP₁がヘルスチェックリスト22を受信するプロセッサとなることはないため、ヘルスチェックリスト22の巡回には影響がない。この方法は修正が最小限になるという特徴がある。

【0086】一方、図12の(3)のB方式はヘルスチェックリスト22よりプロセッサP₁を除いてインデックスの数を1個減らす方法である。この方法では図12の(1)におけるインデックス“3”の送信先プロセッサ番号をP₁

してインデックス“5”以降を順次繰上げた形となっている。この方法は、障害プロセッサ P_i がヘルスチェックリスト22から除かれるとともに、巡回するプロセッサの数が明確になると言う特徴がある。

【0087】以上の処理が行われたのち、プロセッサ P_i から修正された通常モードのヘルスチェックリスト22が送信される。このヘルスチェックリスト22はプロセッサ P_i まで前回と同じく巡回されるが、プロセッサ P_i がヘルスチェックリスト22を読み取ると送信先プロセッサとしてプロセッサ P_i が指定されているため、ヘルス

チェックリスト22はプロセッサ P_i の代わりにプロセッサ P_j に送られ、以後、プロセッサ P_j 、 P_k を経てプロセッサ P_i に返送される。

【0088】次に、プロセッサ P_i の障害が回復し、再びシステムに組み込まれる場合について説明する。図17はプロセッサ P_i を組み込む前のヘルスチェックリスト22の巡回経路を示している。図に実線で示すように、通常モードのヘルスチェックリスト22はプロセッサ P_i を通らずに巡回されているが、この状態でプロセッサ P_i の障害が回復し、システムに組み込める状態になると、

プロセッサ P_i より他の全プロセッサ $P_1 \sim P_3$ 、 $P_5 \sim P_6$ に対して組み込み通知が行われる（図16に点線で示す）。

【0089】組み込み通知を受けた各プロセッサ $P_1 \sim P_3$ 、 $P_5 \sim P_6$ はそれぞれ自プロセッサ内のプロセッサ状態管理テーブル21のプロセッサ P_i の状態を「非稼働」から「稼働中」に変更する。これによって、全プロセッサ $P_1 \sim P_6$ のプロセッサ状態管理テーブル21は図3の(2)の状態から同図(1)の状態に戻る。

【0090】また、組み込み通知を受信したときに巡回中の通常モードのヘルスチェックリスト22を受信しているプロセッサはそのヘルスチェックリスト22を送信する前にヘルスチェックリスト22を図12の(2)または(3)から(1)のように修正して次の送信先プロセッサに送信する。これによって、このヘルスチェックリスト22は次にプロセッサ P_j に巡回されたときにプロセッサ P_i ではなくプロセッサ P_j に対して送信されるようになる。

【0091】図17は以上の組み込み通知が送られる経路を図示しており、図20は組み込み通知が行われる前後のヘルスチェックリスト22と組み込み通知の送信シーケンスを図示している。図20ではプロセッサ P_i が通常モードのヘルスチェックリスト22を受信した状態で組み込み通知が行われ、組み込み通知による処理終了後にプロセッサ P_i からヘルスチェックリスト22の送信が再開される状態を示している。なお、図20における組み込み通知の送信順序は一例を示したものである。

【0092】以上、障害回復後のプロセッサの組み込み方法を説明したが、組み込み方法として次のような別方法（図示省略）を使用することもできる。この別方法で

ッサ $P_1 \sim P_3$ 、 $P_5 \sim P_6$ はプロセッサ状態管理テーブル21を変更したのち、通知元のプロセッサ P_i に対して組み込みが終了したことを知らせる応答信号を返送する。

【0093】プロセッサ P_i は組み込み通知を行った全プロセッサ $P_1 \sim P_3$ 、 $P_5 \sim P_6$ から応答信号を受信すると、自プロセッサの前順位のプロセッサ P_j に対してヘルスチェックリスト22への組み込み依頼を行う。前順位のプロセッサ P_j はこの依頼を受けたのち、ヘルスチェックリスト22（このとき受信するヘルスチェックリスト22はプロセッサ P_i が除かれた図12の(2)または(3)の内容となっている）を受信するとそのヘルスチェックリスト22を図12の(1)の内容に修正し、送信先プロセッサとして新たに指定されたプロセッサ P_i に対してそのヘルスチェックリスト22を送信する。この別方法は情報の送受信回数は増加するが組み込みの完了を確認してヘルスチェックリスト22の修正を行うため、信頼性が高くなる。

【0094】以上、図3～図20により本発明の実施例を説明したが、図3～図20はあくまで本発明の一例を示したものに過ぎず、本発明が図示されたものに限定されるものでないことは言うまでもない。

【0095】例えば、上記においてはプロセッサに障害が発生したことを前提として説明したが、保守点検や増設などのために非稼働状態のプロセッサが発生したときにも本発明が適用できることは明らかである。

【0096】また、図10においてはヘルスチェック信号を送信する相手のプロセッサを確認するためにカウンタを用い、カウンタ値 x と自プロセッサの番号 k をインデックスとして図3のプロセッサ状態管理テーブル11から直接送信先プロセッサの番号 $P_{k,x}$ を得ていたが、次の何れかの方法を用いることも可能である。

【0097】第1の方法は、プロセッサ状態管理テーブル11に自プロセッサのデータが記憶されている箇所のインデックス（またはアドレスでもよい）を予め知っておき、カウンタを用いずにプロセッサ状態管理テーブル11にアクセスする方法である。この方法ではそのインデックスを用いてプロセッサ状態管理テーブル11の自プロセッサのデータを読み出す。例えば、プロセッサ P_i の例では自プロセッサのデータが記憶されているインデックス“1”にアクセスすると自プロセッサのヘルスチェック信号送出順位が“1”であることが確認できるので、次にヘルスチェック信号送出順位が“2”であるプロセッサを検索してプロセッサ P_j を得ることができる。

【0098】第2の方法はインデックスを使用せずに直接プロセッサ状態管理テーブル11にアクセスし、プロセッサ番号の中から自プロセッサ番号を見つける。例えばプロセッサ P_i の場合、プロセッサ番号の項で P_i を検出できれば、以後は第1の方法と同様にしてヘルスチェ

の方法を用いても本発明の効果が変わらないことは明らかである。

【0099】また、図5及び図8の障害通知、図6及び図9の組み込み通知、図16の障害通知、図17及び図20の組み込み通知については、通知を受けたプロセッサより応答信号を返すようになっていないが、応答信号を返すようにしても本発明の効果は変わらない。

【0100】また、以上の説明においては、巡回型のプロセッサ障害検出方法におけるプロセッサ状態管理テーブル21は応答監視型のプロセッサ障害検出方法におけるプロセッサ状態管理テーブル11と同一内容であるとしたが、巡回型のプロセッサ障害検出方法においてはヘルスチェックリスト22に送信先のプロセッサ番号が指定されるので、図3に示すプロセッサ状態管理テーブルのヘルスチェックリスト送出順位を削除し、プロセッサ番号とプロセッサ状態のみで構成し、プロセッサ状態管理テーブル21では各プロセッサが稼働中であるか非稼働であるかを確認するのみとすることもできる。この場合も本発明の効果は変わらない。

【0101】

【発明の効果】以上説明したように、本発明によれば、バスによって接続された複数のプロセッサが互いにプロセッサ障害を検出するシステムにおいて、個々のプロセッサが障害の有無を確認するための情報を送受信する相手プロセッサがそれぞれ1台に限られるため、各プロセッサが他の全プロセッサとの間で障害確認のための情報を送受信するメッシュ状の相互監視型障害検出方法に比して各プロセッサの障害検出のための処理負担が著しく減少する。また、プロセッサの数が増加しても各プロセッサの障害検出のための処理が大きく増加することがないため、システム全体の処理能力を大きく低下させることがない。

【0102】以上により、本発明はバスに接続された複数の処理プロセッサが分担して処理を行うシステムにおけるプロセッサ障害検出の効率化と、かかるシステム、特にプロセッサの数が多きシステムの処理能力の向上に大きく貢献する。

【図面の簡単な説明】

【図1】 本発明の原理説明図(1)

【図2】 本発明の原理説明図(2)

【図3】 本発明の実施例プロセッサ状態管理テーブル

構成説明図

【図4】 本発明の実施例情報送受信経路説明図(応答監視型-1)

【図5】 本発明の実施例情報送受信経路説明図(応答監視型-2)

【図6】 本発明の実施例情報送受信経路説明図(応答監視型-3)

【図7】 本発明の実施例情報送受信シーケンス図(応答監視型-1)

10 【図8】 本発明の実施例情報送受信シーケンス図(応答監視型-2)

【図9】 本発明の実施例情報送受信シーケンス図(応答監視型-3)

【図10】 本発明の実施例動作フロー図(応答監視型-1)

【図11】 本発明の実施例動作フロー図(応答監視型-2)

【図12】 本発明の実施例ヘルスチェックリスト構成説明図(1)

20 【図13】 本発明の実施例ヘルスチェックリスト構成説明図(2)

【図14】 本発明の実施例情報送受信経路説明図(巡回型-1)

【図15】 本発明の実施例情報送受信経路説明図(巡回型-2)

【図16】 本発明の実施例情報送受信経路説明図(巡回型-3)

【図17】 本発明の実施例情報送受信経路説明図(巡回型-4)

30 【図18】 実施例情報送受信シーケンス図(巡回型-1)

【図19】 実施例情報送受信シーケンス図(巡回型-2)

【図20】 実施例情報送受信シーケンス図(巡回型-3)

【図21】 システム構成図

【図22】 従来技術のプロセッサ障害検出方法説明図

【符号の説明】

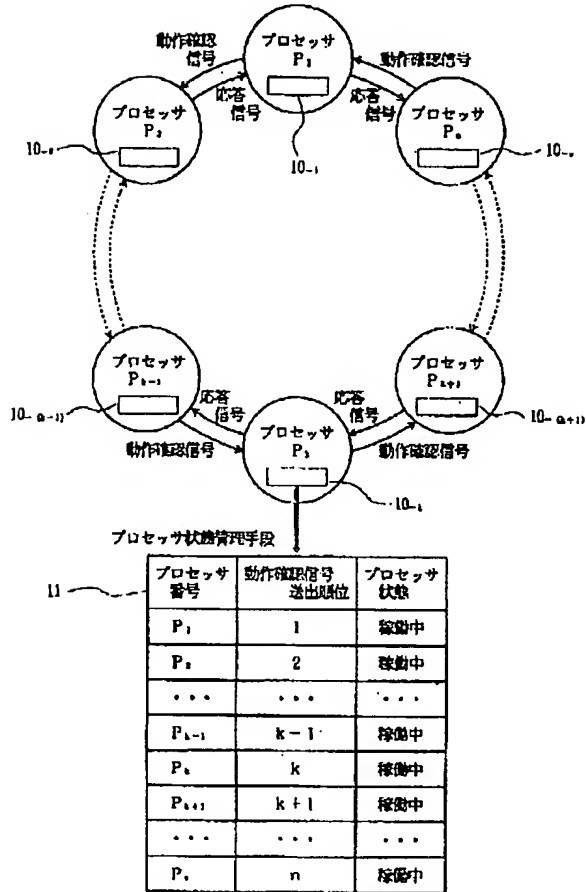
10、10_a、20、20_a プロセッサ

40 11、21 プロセッサ状態管理手段

22 動作確認情報

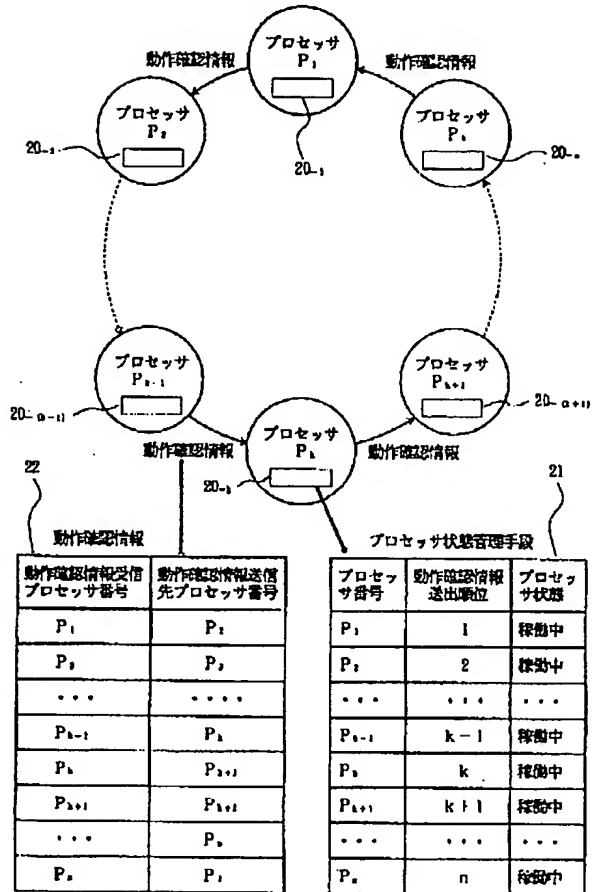
【図 1】

本発明の原理説明図 (1)



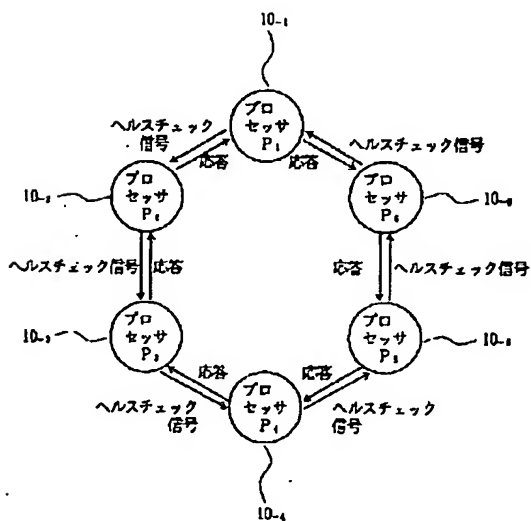
【図 2】

本発明の原理説明図 (2)



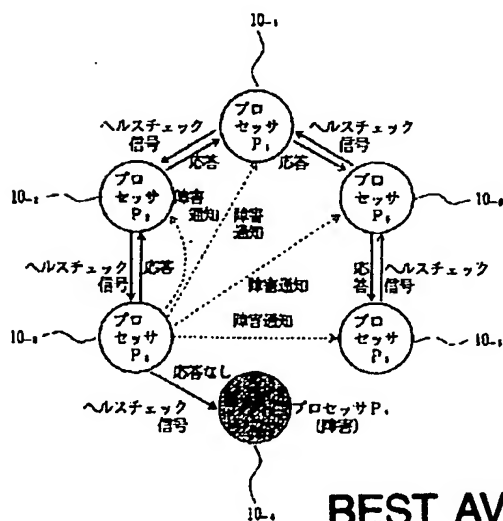
【図 4】

本発明の実施例情報送受信経路説明図 (応答監視型-1)



【図 5】

本発明の実施例情報送受信経路説明図 (応答監視型-2)



【図 3】

本発明の実施例プロセッサ状態管理テーブル構成説明図

(1) 正常状態

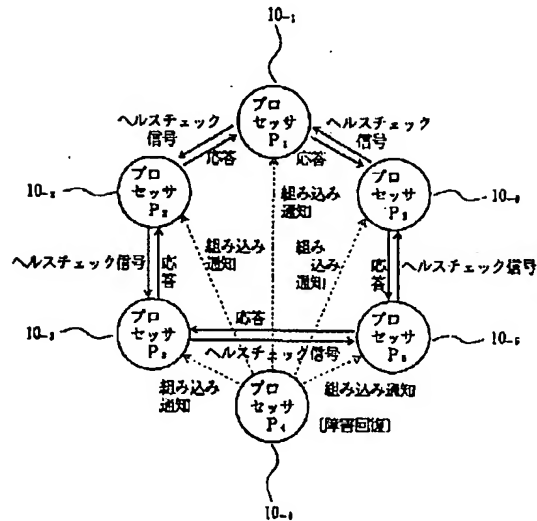
インデックス	プロセッサ 番号	ヘルスチェック信号/ ヘルスチェックリスト 送出順位	プロセッサ 状態
1	P ₁	1	稼働中
2	P ₂	2	稼働中
3	P ₃	3	稼働中
4	P ₄	4	稼働中
5	P ₅	5	稼働中
6	P ₆	6	稼働中

(2) プロセッサ P₄ 非稼働状態

インデックス	プロセッサ 番号	ヘルスチェック信号/ ヘルスチェックリスト 送出順位	プロセッサ 状態
1	P ₁	1	稼働中
2	P ₂	2	稼働中
3	P ₃	3	稼働中
4	P ₄	4	非稼働 （故障発生）
5	P ₅	5	稼働中
6	P ₆	6	稼働中

【図 6】

本発明の実施例情報送受信経路説明図（応答監視型-3）

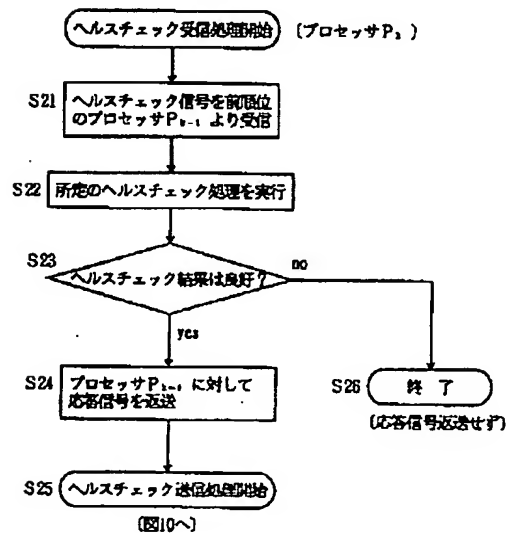
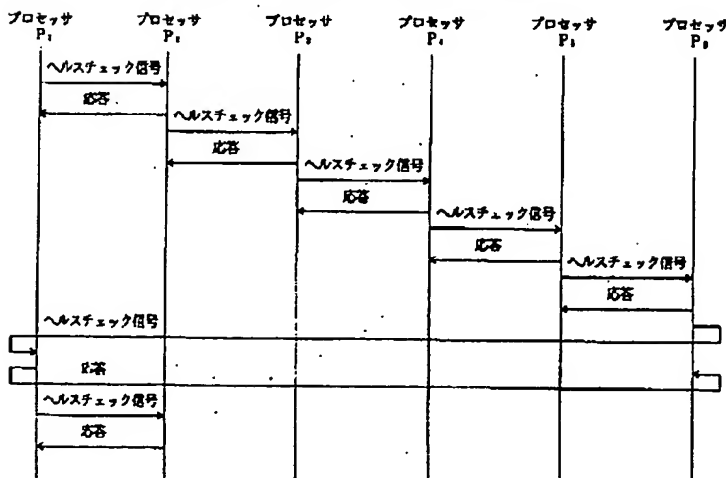


【図 11】

本発明の実施例動作フロー図（応答監視型-2）

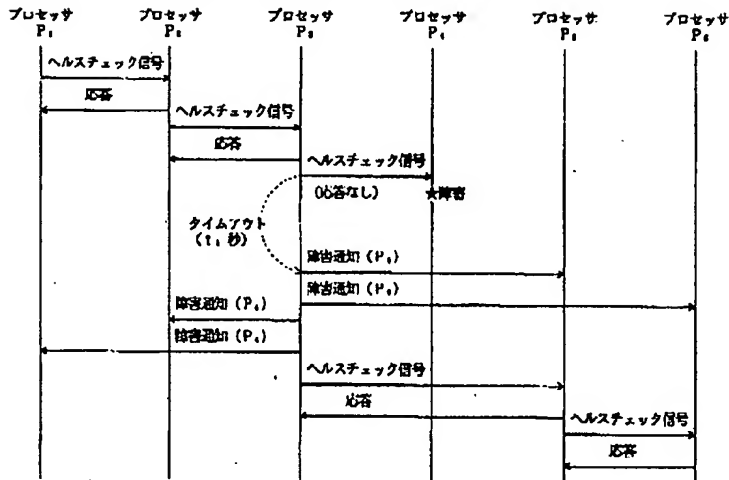
【図 7】

本発明の実施例情報送受信シーケンス図（応答監視型-1）



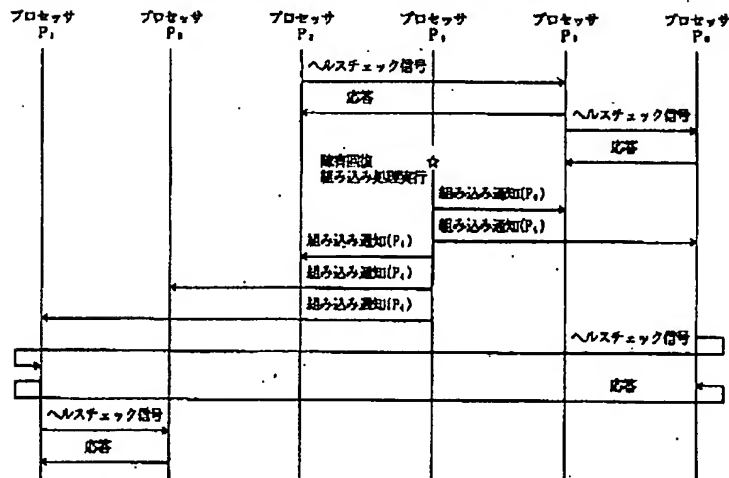
【凶8】

本発明の実施例情報送受信シーケンス図（本発明形態-2）



【図 9】

本発明の実施例は特許請求の範囲（図面参照）に示す通りである。



【图 1.3】

本発明の実施例ヘルスチェックリスト構成説明図（２）

(1) 障害モード:

インデックス	モード	受信プロセッサ番号	送信先プロセッサ番号
1	障害	P ₁	P ₁
2	障害	P ₂	P ₁
3	障害	P ₃	P ₁
4	障害	P ₄	P ₁
5	障害	P ₅	P ₁
6	障害	P ₆	P ₁

【図 12】

本発明の実施例ヘルスチェックリスト構成説明図（１）

・(1) 通常モード (全プロセッサ稼働中)

インデックス	モード	受信プロセッサ番号	送信先プロセッサ番号
1	通常	P ₁	P ₁
2	通常	P ₁	P ₃
3	通常	P ₁	P ₁
4	通常	P ₁	P ₃
5	通常	P ₂	P ₁
6	通常	P ₂	P ₁

(2) 通常モード (プロセッサ P、非稼働状態 - A 方式)

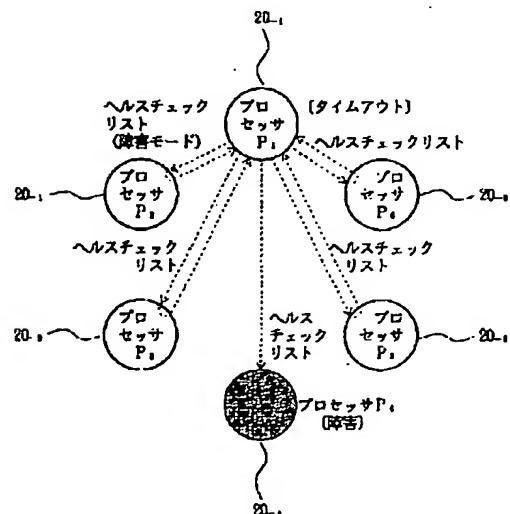
インデックス	モード	受信プロセッサ番号	送信先プロセッサ番号
1	通常	P ₁	P ₁
2	通常	P ₁	P ₂
3	通常	P ₁	P ₃
4	通常	P ₁	P ₄
5	通常	P ₁	P ₅
6	通常	P ₁	P ₁

(9) 通常モード (プロセッサ P、非降倒状態-B方式)

インデックス	モード	受信プロセッサ番号	送信先プロセッサ番号
1	通常	P ₁	P ₂
2	通常	P ₁	P ₃
3	通常	P ₁	P ₃
4	通常	P ₂	P ₆
5	通常	P ₂	P ₁

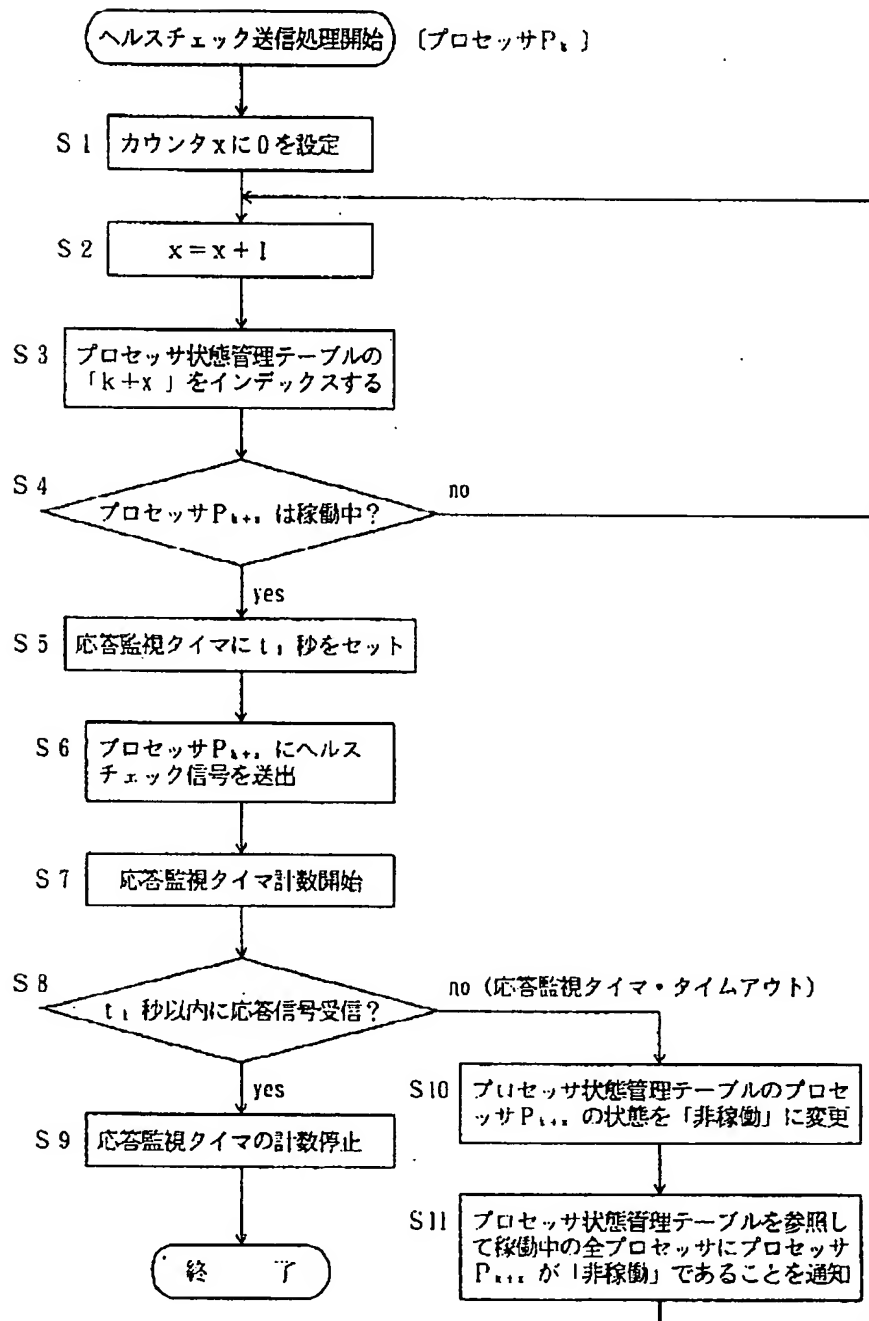
【图 15】

本発明の実施例情報送受信回路説明図（巡回型-2）



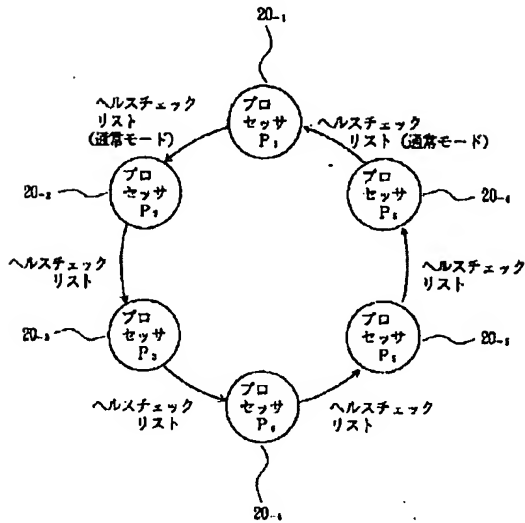
【図 10】

本発明の実施例動作フロー図（応答監視型-1）



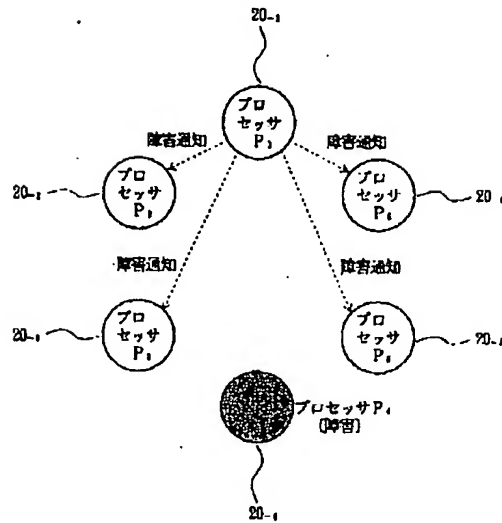
【図 1 4】

本発明の実施例情報送受信経路説明図 (巡回型 - 1)



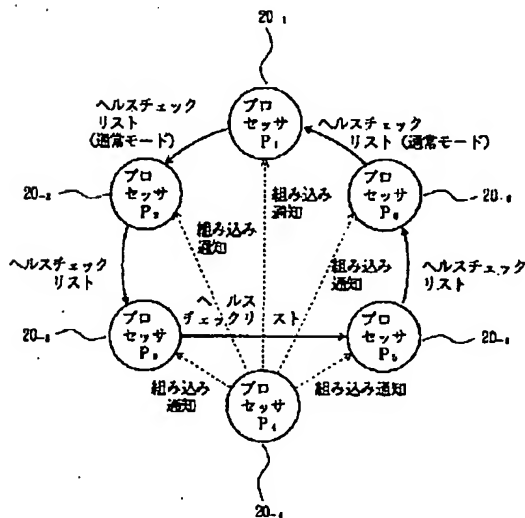
【図 1 6】

本発明の実施例情報送受信経路説明図 (巡回型 - 3)



【図 1 7】

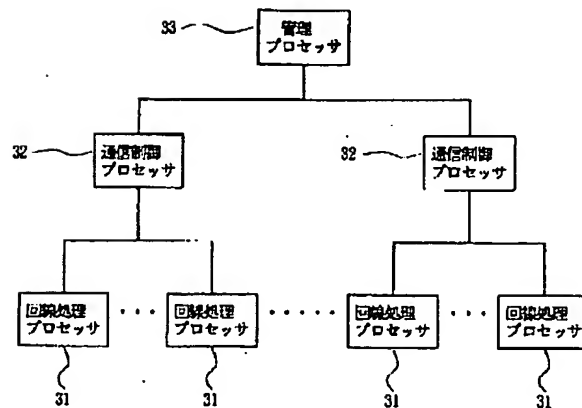
本発明の実施例情報送受信経路説明図 (巡回型 - 4)



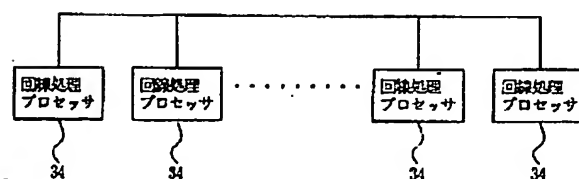
【図 2 1】

システム構成図

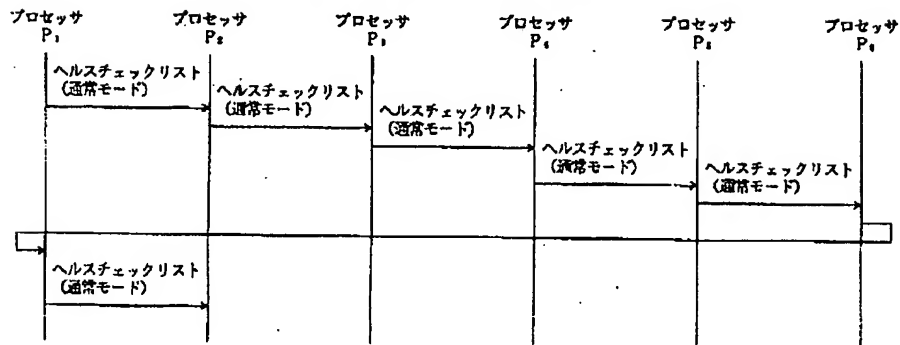
(1) 階層構造システム



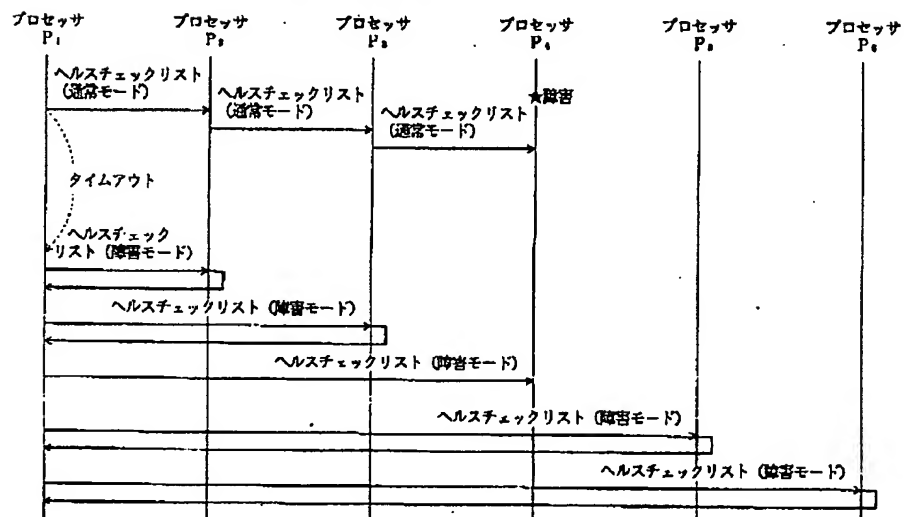
(2) 非階層構造システム



本発明の実施例情報送受信シーケンス図（図回型－１）

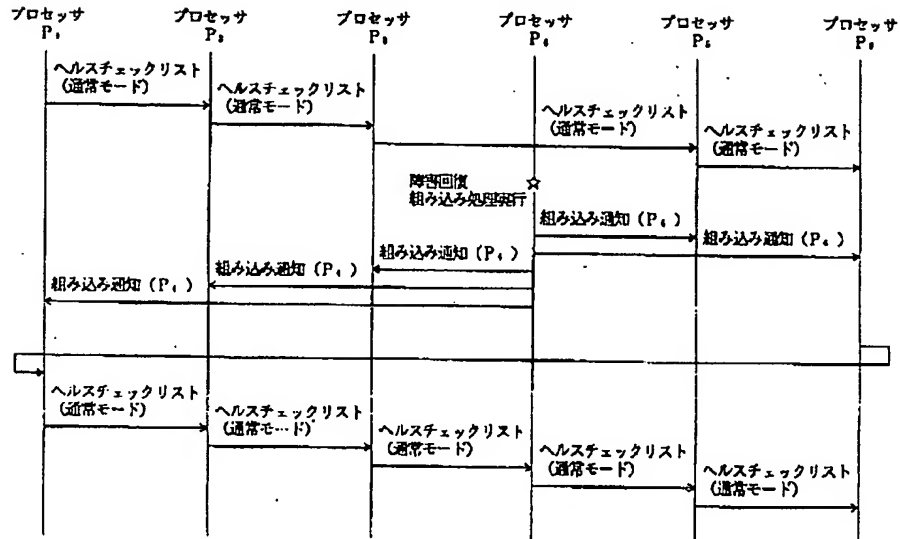


本発明の実施例情報送受信シーケンス図（図回型-2）



【図 20】

本発明の実施例情報送受信シーケンス図 (図 20-3)



【図 22】

従来技術のプロセッサ障害検出方法説明図

